



Driver fatigue detection method based on multi-feature empirical fusion model

Yanbin Qin¹ · Hongming Lyu¹ · Kaibin Zhu¹

Received: 8 September 2023 / Revised: 10 July 2024 / Accepted: 18 August 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

As the number of long-distance commuters continues to rise, driver fatigue has become a major contributor to traffic accidents, underscoring the critical need for real-time fatigue detection and prevention. Facial expressions serve as crucial indicators that directly reflect the driver's fatigue state, which may vary due to individual differences. This paper presents a comprehensive visual monitoring approach for real-time driver fatigue detection. The proposed method uses a vehicle-mounted camera to extract the driver's facial features through an empirical multi-feature fusion model. By determining appropriate thresholds based on individual driving habits and conditions, the algorithm maps multi-dimensional facial behaviors to corresponding Karolinska Sleepiness Scale (KSS) scores and fatigue levels, accurately identifying four states: awake, mild fatigue, moderate fatigue, and severe fatigue. Evaluated on a dataset of 2,555 validated samples obtained from real-world driving conditions, the method demonstrated an impressive average accuracy of 98.35% during both training and experimentation. Furthermore, a comparative analysis against state-of-the-art fatigue detection algorithms on a self-curated dataset revealed that the proposed approach achieved the highest average accuracy of 98.9%, with lower computational requirements and a lighter weight. The mean average precision (mAP) was 1.6% higher than the lightweight Efficient Det-D2, while also having fewer model parameters and reduced computational complexity.

Keywords Fatigue detection · Deep learning · Decision-making · Human behavior analysis

✉ Hongming Lyu
lhmyg@163.com

Yanbin Qin
qinyanbin1@aliyun.com

Kaibin Zhu
Zkb1631@163.com

¹ School of Automotive Engineering, Yancheng Institute of Technology, Yancheng 224051, China

1 Introduction

The importance of road traffic crashes as a global social problem cannot be overstated. Statistics show that fatigued driving is the primary cause of over 60% of traffic accidents [1, 2]. Fatigued driving occurs when drivers experience physiological and psychological dysfunction due to prolonged driving that significantly impairs their driving ability [3–5]. This condition affects several facets of the driver's cognitive processes, including attention, perception, reasoning, judgment, willpower, decision making, and reaction time. Driving while fatigued can significantly increase the risk of road crashes, highlighting the critical role of early detection of this condition in ensuring road safety.

Drivers exhibit distinct physiological and psychological symptoms when they become fatigued [6]. According to the length of driving time, fatigue driving can be classified as either short-term or long-term. In the case of short-term fatigue driving, drivers exhibit the following characteristics: (1) Increased frequency of blinking, fatigue, and reduced attention to safety. (2) Inaccurate and untimely gear shifting and lack of focused attention. (3) Inability to adjust driving behavior, such as acceleration, deceleration, and steering, based on changing road conditions. Meanwhile, long-term fatigue driving is characterized by: (1) Dry mouth, frequent yawning, head nodding, and difficulty keeping the head upright. (2) Painful, dry eyes that open and close, drowsiness, and blurred vision. (3) Low mood, slow reaction time, and impaired judgment [7–9].

At present, research on fatigue driving is mainly focused on the following three directions:

One is to evaluate and detect fatigue driving state based on signal characteristics. This approach mainly includes the use of electrocardiogram (ECG) [10, 11] and photoplethysmography (PPG) to detect ECG signals [12, 13], multi-wave electroencephalography (EEG) signals [14], surface electromyography (sEMG) to detect electroencephalography (EMG) signals [15], and measurement of electrooculography (EoG) signals between the cornea (with positive electricity) and the retina (with negative electricity) [16]. These methods of detecting fatigue using various physiological signals of the human body have strong theoretical support in biology and can achieve high detection accuracy. However, they require drivers to wear special detection instruments during the measurement process, which can greatly affect their ability to drive. In addition, the cost of professional detection instruments is generally very high, which makes it difficult to apply in practice.

The second is driving fatigue detection based on vehicle characteristics. It mainly relies on indirectly detecting and judging signs of fatigue through vehicle behavior, such as steering wheel angle, driving speed, acceleration, trajectory, lane offset, pressure exerted on the seat by various parts of the driver's body, and brake pedal pressure [17–19]. This vehicle and driver behavior-based method of fatigue detection avoids direct physical contact with the driver and does not interfere with the driver's driving behavior. However, factors such as weather and road conditions, vehicle models, and driver habits can significantly affect the accuracy of these detection methods, making them relatively less robust than other approaches.

The third is facial fatigue detection. It is performed by eyes, mouth, expression, nose and head posture [20–22]. Compared with traditional approaches, computer vision-based facial feature fatigue detection has several advantages, including non-contact and non-interference operation and high detection accuracy, making it a new research hotspot in this area. Mbouna [9] used SVM to classify both alert and non-alert states. Zhao C [23] identified fatigue expressions, extracted fatigue expression features, and classified them

by the stochastic subspace integration model of SVM with a polynomial kernel function. Ahmad [24] mainly studied eye opening and closing with head movement detection to detect fatigue. They used the "Viola-Jones" method for face detection and the CART method to detect the Haar features of the human eye ROI region. Ghoddoosian [25] used dlib to extract human eye key points, compute EAR (Eye Aspect Ratio) values, and extract blink features; they also defined a time window to transfer blink features to a HM-LSTM network, learned the characteristics of blinks in the time dimension, and then used a combination of fully connected layers, regression units, and discrete values to map KSS values to three types of error states. Li [26] detects driver fatigue by measuring the duration of eye closure, blink frequency, and yawn frequency. For face detection, the dlib toolkit of YOLOv3 Tiny is used to extract feature vectors of the eye and mouth, and then an SVM classifier evaluates the fatigue state based on the characteristics of eye closure time, blink frequency, and yawn frequency.

The remainder of this paper is organized as follows: Section 2 describes the empirical fusion of KSS values of multiple fatigue behaviors using f_1 , f_2 , and f_3 operators to establish the logical relationship of multiple fatigue behaviors and two KNN models used for real-time early fatigue and post-fatigue estimation. Section 3 details the detection process and the determination of the KSS values. Section 4 discusses the experiment and the results. The authors used a self-curated dataset for analysis and processing, and a simulation of driving fatigue on a real vehicle, which gave promising results compared to other algorithms. Finally, Section 5 presents some concluding remarks.

2 Facial fatigue detection algorithm

There are several challenges associated with the practical application of fatigue detection technology. One such challenge is the requirement for real-time performance, which limits the available models. While deep learning models offer high accuracy, they are time-consuming during the fatigue inference phase, making it difficult to study fatigue detection from a modeling perspective. Fatigue detection is also a classification task, but compared to object classification, its boundaries are not well defined. Early fatigue testing typically focuses on one specific fatigue behavior, and building a multi-feature model requires unsupervised or supervised learning methods. However, the construction of multiple fatigue behavior features can be time-consuming, so the fusion analysis model should not be used for extended periods of time. In addition, KSS labeling involves many subjective factors, which can lead to overfitting when training supervised models.

The method proposed in this paper is a visual monitoring technique that analyzes the driver's facial features in real time using a vehicle camera to determine the driver's fatigue state. The method uses a multi-feature empirical fusion model that considers the driver's personal situations and habits by determining an appropriate threshold. The model assigns a Karolinska Sleepiness Scale (KSS) score and a fatigue behavior weight that maps the multidimensional facial behavior combination to a fatigue-related KSS score to finally assess the driver's fatigue state. The authors believe that this approach provides an effective and non-invasive way to determine driver fatigue status, which can be beneficial for the development of advanced driver assistance systems and the reduction of driver-related accidents.

Figure 1 shows the fatigue detection framework, which includes three operators (f_1 , f_2 , and f_3) that establish the logical relationship between different fatigue behaviors by

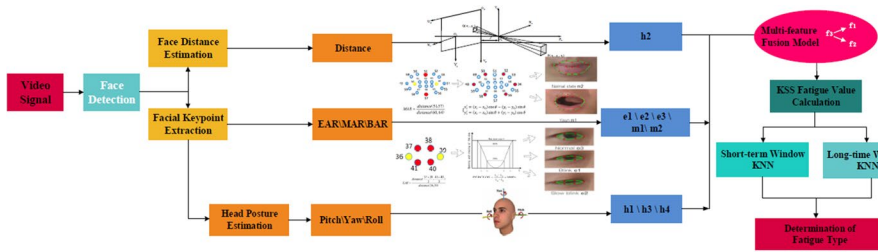


Fig. 1 Detection framework of the facial fatigue detection algorithm

empirically fusing the KSS values. In addition, the framework includes two K-Nearest Neighbors (KNN) models—one for short-term fatigue detection and another for long-term fatigue detection. The use of both models ensures that the proposed method can detect both early and late fatigue in real time, making it applicable to real-world driving scenarios. The overall architecture of the framework represents a comprehensive and practical approach to driver fatigue detection.

The steps of the fatigue detection algorithm are as follows:

1. Face detection: use the SCRFD-0.5GF+ model.
2. 68 face key point detection: use MobileNetV3-56++ model.
3. Head motion detection: use the EPNP algorithm to calculate 3 rotational degrees of freedom and 3 translational degrees of freedom of the head posture. The first-order difference and threshold comparison of each degree of freedom are calculated to detect nodding, normal movement, head rest and head tilt forward and backward.
4. Head forward and backward motion detection: use the pinhole imaging principle to calculate the distance between the face and the camera. First-order distance difference and threshold judgment are used to detect the forward and backward tilt motion.
5. Blink detection: adaptive blink threshold of head posture based on calibrated EAR_ EAR_Threshold, using EAR and PERCLOS for two-stage blink detection.
6. Yawn detection: A yawn detection algorithm based on head posture uses MAR (Mouth Aspect Ratio) and FOM (Frequency of Occurrence of Mouth Opening) for two-stage yawn detection.

Table 1 shows the fatigue behavior code and KSS value setting used in the proposed algorithm. The table lists three types of detection features, which include the yawning state (m1) and normal state (m2) of the mouth, fast blinking (e1), slow blinking (e2), and normal state (e3) of the eye, and finally, the nodding behavior (h1), leaning forward and backward behavior

Table 1 Fatigue behavior codes

Position	Fatigue behavior			
Mouth	Yawn m1	Normal m2	/	/
Eyes	Blink e1	Slow blink e2	Normal e3	/
Head	Nod h1	Tilt forward and backward h2	Normal movement h3	Static h4

Table 2 Karolinska Sleepiness Quantification Table [27]

Verbal description	KSS Score
Extremely alert	1
Very alert	2
Alert	3
Rather Alert	4
Neither alert nor sleepy	5
Some signs of sleepiness	6
Sleepy, but no effort to keep awake	7
Sleepy, some effort to keep awake	8
Very sleepy, great effort to keep awake, fighting sleep	9

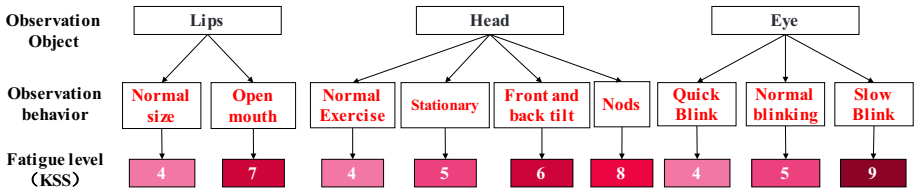


Fig. 2 Relationship between object, behavior, and KSS

(h2), normal movement behavior (h3), and static state (h4) of the head posture. These behaviors are used to identify the instances of driver fatigue and map them to the corresponding KSS values. Thus, the table provides a comprehensive understanding of the driver’s fatigue state and enables effective fatigue detection.

Given the objective of only detecting the driver’s fatigue status, the proposed algorithm adopts a focused approach that reduces computational burden and increases efficiency. Specifically, in order to optimize resource utilization, we use a subset of the KSS sleepiness quantification table, specifically values ranging from 4–9, to assess driver fatigue, as detailed in Table 2.

There is a corresponding relationship between the observed object, the observed behavior and the value of the fatigue level (KSS). Specifically, the mouth fatigue range is 4 to 7, the head fatigue range is 4 to 8, and the eye fatigue range is 4 to 9, as shown in Fig. 2.

The empirical fusion of multiple fatigue behavior KSS values is to use normalized empirical KSS values and the number of fatigue behavior detection times, as shown in Fig. 3. Three operators are defined: singleton (f1), mutual (f2), and active/inhibit (f3). The cause-and-effect diagram of various fatigue behaviors constructed by human experience gives specific meanings to the three operators:

$$f_1 = \alpha \times KSS_{norm_{code_i}} \times count_{norm_{code_i}} \tag{1}$$

$$f_2 = \tanh\left(\beta\left(\sum_j \left(KSS_{norm_{code_j}} \times count_{norm_{code_j}}\right)\right)\right) + \alpha Max_j(KSS_{norm_{code_j}} \times count_{norm_{code_j}}) \tag{2}$$

$$f_3 = \tanh\left(\beta\left(\sum_k \left(KSS_{norm_{code_k}} \times count_{norm_{code_k}}\right)\right)\right) \tag{3}$$

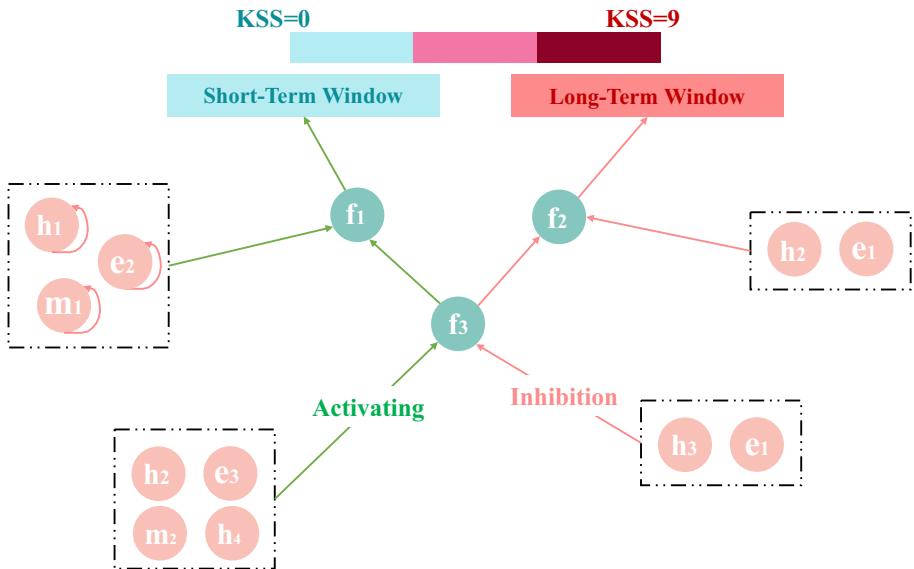


Fig. 3 KSS value fusion diagram of multiple fatigue behaviors

$$\text{activate} = f_3 = -\text{inhibit} \quad (4)$$

The f_1 operator is designed to detect three common signs of fatigue: blinking, yawning, and nodding. First, a high KSS value is assigned to determine the onset of fatigue, and the operator then calculates the frequency of these signs to estimate the level of subsequent fatigue.

The f_2 operator focuses on identifying early fatigue signs such as head tilting forward/backward and rapid blinking. To calculate the maximum level of early fatigue, the operator applies the mean KSS value assigned at the beginning and performs a count, followed by the use of the tanh activation function and the max function.

The f_3 operator plays a complementary role to the f_1 and f_2 operators. It not only triggers the f_1 operator and amplifies the subsequent fatigue value, but also dampens the f_2 operator to reduce early fatigue values and mitigate potential early fatigue misjudgments.

The facial fatigue detection algorithm uses long and short term KNN to learn the fatigue threshold, and takes both short window KSS and long window KSS extracted from each video as training samples for two separate KNN models. To ensure efficient performance, the dataset is pre-processed and normalized for real-time use during early and late fatigue estimation.

3 Facial feature point detection

The KSS values are determined in several steps. First, face detection is performed using the SCRFD-0.5GF+ algorithm. Next, the key points of 68 faces are detected using the MobileNetV3-56++ algorithm. Subsequently, the EPNP algorithm is used to calculate three rotational degrees of freedom and three translational degrees of freedom of head movements. The pinhole imaging principle is applied to detect whether the head is moving forward or

backward. Finally, two-stage slow-blink detection is performed using the EAR and PER-CLOS (Percentage of Eye Closure over Time). Similarly, two-step yawn detection is performed using the MAR and FOM measures. These steps help to accurately quantify the KSS values during fatigue detection.

3.1 Face detection

The face detection method used in this paper is SCRFD-0.5GF+ [27]. It is a lightweight model that is well suited for deployment on edge devices with limited computational resources due to its small size and low computational cost. SCRFD-0.5GF+ uses the backbone network to extract features from the input image, and predicts the position and category of objects in the image through a series of convolutional layers. A feature pyramid network (FPN) is used to capture multi-scale features. The FPN architecture is a bottom-up and top-down approach that aggregates feature maps from different levels of the backbone network to enable the model to effectively detect targets of different sizes and scales. The training samples are randomly cut into square patches on the backbone network, and more training samples are allocated to smaller scales to improve the detection results via a sample and computation allocation mechanism, as shown in Fig. 4. These output values are used in subsequent steps of feature extraction and KSS value prediction. The "class" represents the category of detected facial features of the driver, including four states such as awake, mildly fatigued, fatigued, and severely fatigued. The "box" represents the region of interest (ROI) detected in the driver's face, i.e. the driver's facial region. The "mask" represents the result of further segmentation and localization of the detected facial ROI, which is used to accurately extract facial features.

Various methods are used to test the accuracy and efficiency of the verification dataset. The test images have a size of 640×640 and are evaluated using FaceBoxes (UCB17),

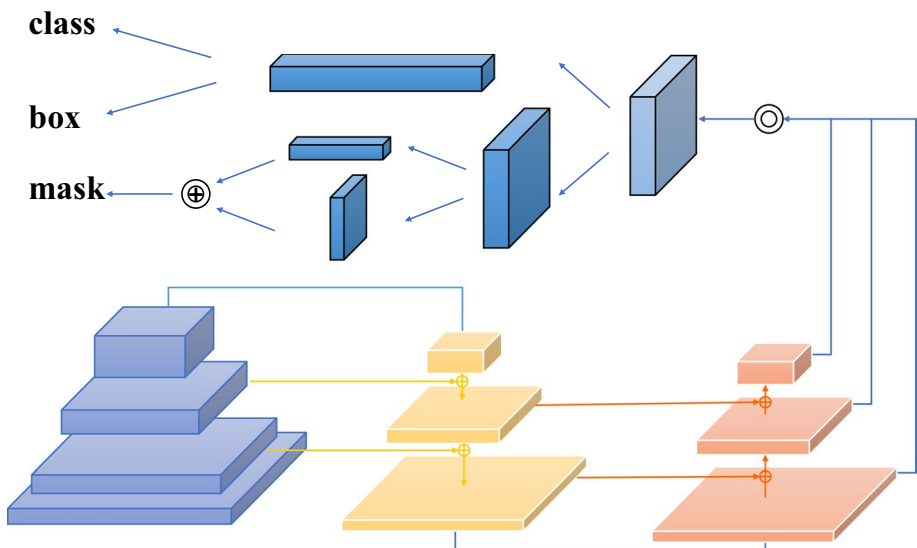


Fig. 4 SCRFD-0.5GF+ Backbone Network Architecture

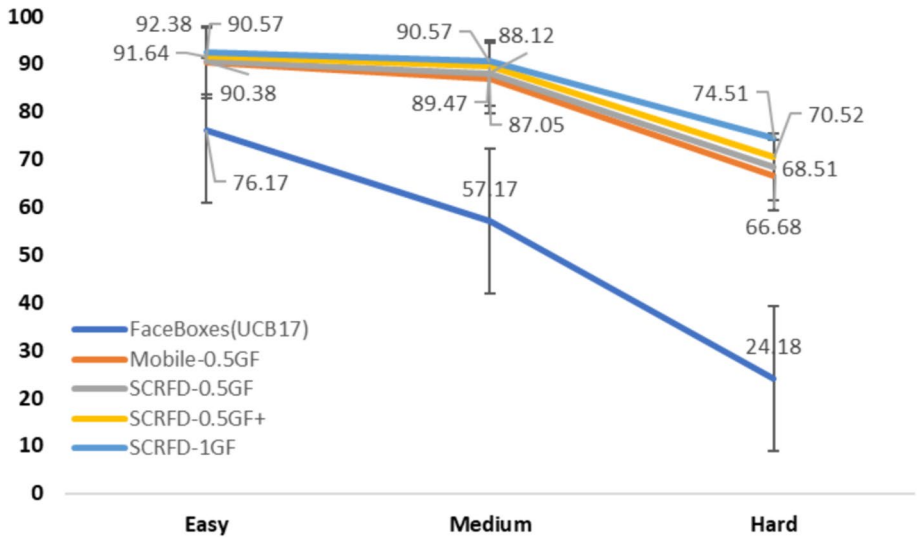


Fig. 5 The accuracy of different methods for testing the validation set

Table 3 Comparison between SCRFD-0.5GF+ and other network structures [28]

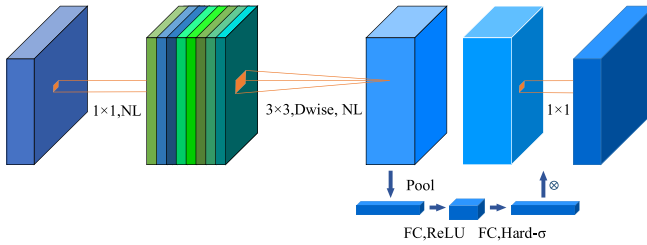
Method	Backbone	Easy	Medium	Hard	#Params (M)	#Flops (G)	Infer (ms)
FaceBoxes(UCB17)	-	76.17	57.17	24.18	1.01	0.275	2.5
Mobile-0.5GF	MobileNet0.25	90.38	87.05	66.68	0.37	0.507	3.7
SCRFD-0.5GF	Depth-wise Conv	90.57	88.12	68.51	0.57	0.508	3.6
SCRFD-0.5GF+	SCRFD-0.5GF+	91.64	89.47	70.52	0.58	0.601	3.6
SCRFD-1GF	Depth-wise Conv	92.38	90.57	74.51	0.64	0.982	4.1

Mobile-0.5GF, SCRFD-0.5GF, and SCRFD-1GF. The "# Params" and "# Flops" represent the number and the product of the parameters, respectively. The 640 images are evaluated on NVIDIA 2080TL \times 640. The test results are shown in Fig. 5 and Table 3.

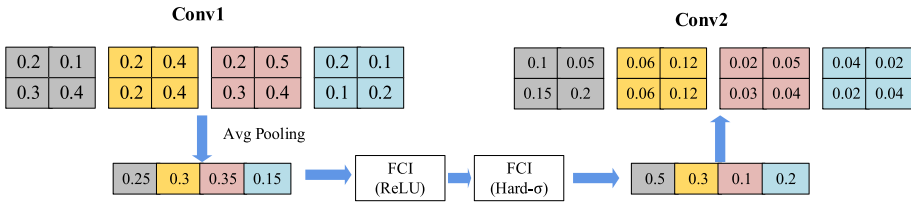
3.2 Detection of Face feature points

After obtaining the face block diagram using the improved SCRFD-0.5GF+ model, the feature points of the face block diagram are detected. For this purpose, the lightweight model MobileNetV3-56++ is used to obtain the face key points. MobileNetV3-56 [29, 30] is a lightweight neural network architecture specifically designed for efficient image classification tasks on mobile devices. An important innovation of MobileNetV3-56 is the use of "squeeze and excite" (SE) blocks, which enhance the capture of channel dependencies and adaptively recalibrate feature maps to improve model accuracy while maintaining a low number of parameters and computational cost. This model can locate key points from coarse to fine with only a few parameters.

The SE module is added to the MobileNetV3 block and the activation function is replaced, as shown in Fig. 6. Since the activation functions used are different, NL



a MobileNetV3-56++ layer



b Squeeze-and-Excite

Fig. 6 MobileNetV3-56 Improvements

(nonlinear) is used in the figure. There are two main types of activation functions: ReLU and Hardswish (Hard- σ). The final 1×1 reduced dimension projection layer uses the linear activation function ($f(x) = x$).

Table 4 shows the architecture of MobileNet V3-56 + +. The Input column indicates the input size, while NBN in the operator indicates the absence of batch normalization. The last conv2d 1×1 layer corresponds to a fully connected layer. Exp size is the dimension

Table 4 Mobile-NetV3-56 + +Body Architecture

Input	Operator	Exp size	#out	SE	NL	s
$224^2 \times 3$	conv2d,3*3	-	16	-	HS	2
$112^2 \times 24$	bneck,3*3	16	16	✓	RE	2
$56^2 \times 24$	bneck,3*3	72	24	-	RE	2
$28^2 \times 24$	bneck,3*3	88	24	-	RE	1
$28^2 \times 40$	bneck,5*5	96	40	✓	HS	1
$14^2 \times 40$	bneck,5*5	240	40	✓	HS	1
$14^2 \times 40$	bneck,5*5	240	40	✓	HS	1
$14^2 \times 40$	bneck,5*5	120	48	✓	HS	1
$14^2 \times 48$	bneck,5*5	144	48	✓	HS	1
$14^2 \times 96$	bneck,5*5	288	96	✓	HS	2
$7^2 \times 96$	bneck,5*5	576	96	✓	HS	1
$7^2 \times 96$	bneck,5*5	576	96	✓	HS	1
$7^2 \times 96$	conv2d,1*1	-	576	✓	HS	1
$7^2 \times 576$	pool,7*7	-	-	✓	HS	7
$1^2 \times 576$	conv2d,1*1	-	1280	-	HS	1
$1^2 \times 1280$	conv2d,1*1	-	k	-	HS	

used by the first conv2d 1×1 layer in the bottleneck for dimension upscaling, and Out is the number of output channels through the bottleneck. SE indicates whether the SE module should be used, and NL indicates which activation function should be used. HS stands for Hardswish, while RE stands for ReLU. Additionally, s is the step size, and when $s = 2$, the length and width become half of the original.

3.3 Mouth feature detection

The mouth feature detection method mainly uses the MobileNetV3-56+ model to capture facial key points, extract mouth feature points, and then identify the shape and motion of the lips. The two-step mouth yawn detection method uses the MAR and FOM methods. MAR [31] indicates the mouth aspect ratio, which is useful for detecting mouth openings. FOM [31] refers to the frame frequency of the open mouth, i.e., the number of times the mouth opens in a given time frame. In the first step, the distance between the upper and lower lips of the mouth is divided by the distance between the left and right lips to obtain the MAR value. Once the MAR value exceeds a certain threshold, it can be preliminarily judged as a yawn. Then, in the second step, the changes in FOM values are counted over a period of time; if the FOM value exceeds a certain threshold, it is considered a yawn. Using MAR and FOM in harmony can improve the accuracy and robustness of yawn detection. Figure 7 illustrates a complete mouth detection process.

3.4 Eye feature detection

Eye feature detection based on calibrated head pose adaptive blink thresholds (adaptive_EAR_threshold) requires two-stage slow blink detection using EAR (Eye Aspect Ratio) and PERCLOS (Percentage of Eye Closure over Time). EAR [32] is typically used to detect whether the eye is closed, which is calculated by measuring the distance between feature points inside the eye, including the position of the eye corners, iris, and tail. When

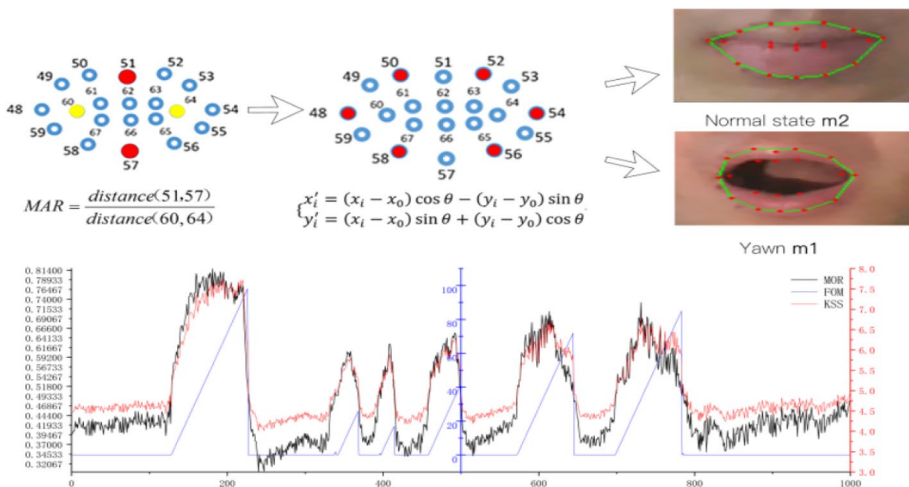


Fig. 7 Mouth feature detection

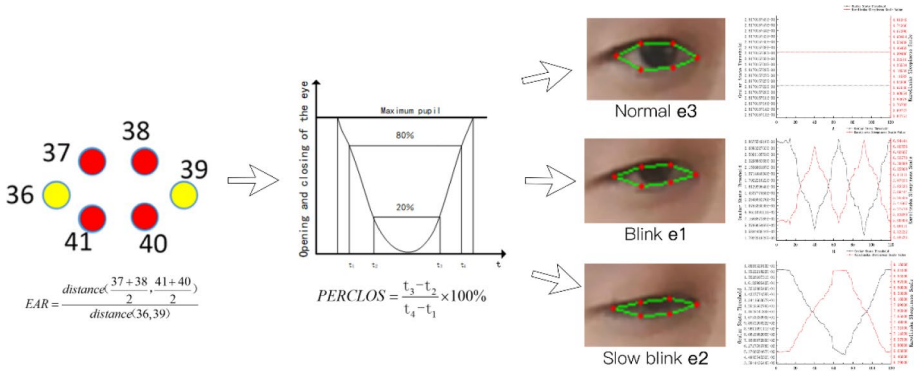


Fig. 8 Eye feature detection

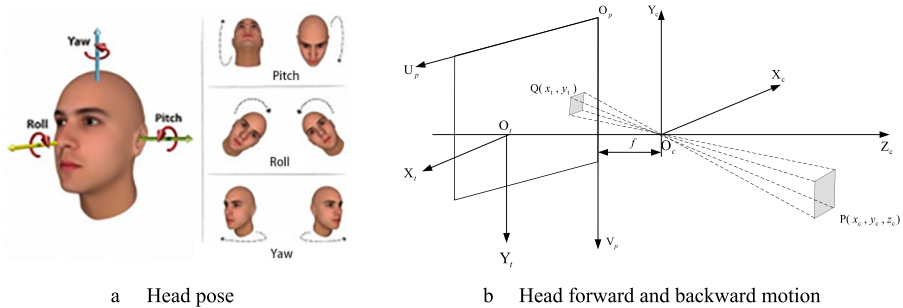


Fig. 9 Head pose feature detection

the eye is closed, the distance between these feature points decreases, resulting in a lower EAR value. PERCLOS [33], often used to evaluate applications such as drowsy driving and crew fatigue, determines the ratio of the time the eye is closed to the total time. The adaptive_EAR_threshold adjusts the EAR threshold based on head pose calibration. Since the shape and position of the eyes can change under different head positions, the EAR threshold must adapt accordingly to ensure accurate detection.

Two-stage slow blink detection divides the closed eye state into two stages: slow blink and fast blink. Slow blinking usually refers to the state in which the eyes are closed for a short time, while fast blinking usually refers to the state in which the eyes are closed for a long time. By dividing the closed state of the eyes into these two stages, the state of the eyes can be more accurately detected and subsequently processed. As shown in Fig. 8, it illustrates a complete eye detection process.

3.5 Head pose feature detection

When SCRFD-0.5GF+ is used to frame the face, head position detection is considered necessary. Head detection can be divided into two parts, as shown in Fig. 9. First, head motion detection is improved by using the EPNP algorithm to calculate the three rotational and three translational degrees of freedom of the head posture. Detection of head nodding, normal head motion, head rest, and forward or backward tilt is done by calculating the

first-order difference and threshold comparison of each degree of freedom. Second, head forward and backward motion detection uses the pinhole imaging principle to calculate the distance between the face and the camera, and then obtains the rate of change of the distance by first-order differentiation. It then determines whether the head is moving forward or backward by comparing the rate of change with the threshold. During the detection phase of head pose estimation, the EPNP algorithm [34] is used to compute the 3 rotational and 3 translational degrees of freedom of the head pose according to known 3D points and corresponding 2D points.

During the head motion detection phase, the first-order difference is computed for each degree of freedom of the head, which gives the rate of change of each degree of freedom. Detection then judges the state of the head motion, including nodding, normal motion, head rest, and forward or backward head tilt, based on a comparison between the change rate and the threshold. By implementing these enhancements, the accuracy and robustness of head pose detection can be improved.

4 Experimental results

The experimental platform mainly consists of the central control unit, camera, horn and bus, and is installed in the experimental vehicle, as shown in Fig. 10.

4.1 Data preparation

Facial feature-based detection was performed using a self-curated hybrid dataset containing four mental state categories: awake, mild fatigue, moderate fatigue and severe fatigue. These four states comprehensively reflect the different stages from full wakefulness to severe fatigue, enabling the study to explore in depth the impact of changes in fatigue level on the performance of the detection algorithm. The data structure is as follows: the network acquisition part provides a total of 1671 sample images, of which 432 are in the awake state, 437 in mild fatigue, 435 in moderate fatigue and 367 in severe fatigue. These images come from different environments and scenarios and show different facial features,



Fig. 10 Construction & environment of the experimental platform

which provide strong support for the generalization ability of the model. Meanwhile, the video database collection part is derived from a 60 fps video stream with a resolution of 780×580 , which ensures the clarity and details of the images. A total of 3104 sample images were collected in this section, including 761 in awake state, 774 in the mild fatigue state, 737 in the moderate fatigue state, and 832 in the severe fatigue state. Finally, 3602 sample images from the public dataset NTHU Drowsy Driving Detection Dataset, Closed Eyes in the Wild (CEW) data were used, of which 834 were in the awake state, 954 in the mild fatigue state, 862 in the moderate fatigue state, and 952 in the severe fatigue state. These images are not only sufficient in number but also of high quality, providing a solid foundation for model training and validation. In total, there are 2027 sample images for the awake, 2165 sample images for mild fatigue, 2034 sample images for moderate fatigue, and 2151 sample images for severe fatigue, as shown in the example of part of the dataset in Fig. 11.

4.2 Experimental analysis

The performance of the algorithm is evaluated by a fivefold cross-validation method and compared with traditional models such as Random Forest (RF), Support Vector Machine (SVM), Radial Basis Function Neural Network (RBF), Bayesian Classification (BC), Random Forest with Multi-feature Fusion (RFFWF) and other models. The RF model uses an SVM-fused random forest algorithm, the SVM model uses the PSO-SVM algorithm, the RBF model uses the SOM algorithm, and the BC model uses a Bayesian model based on PCA. In addition, this study considers the runtime performance of the algorithm, i.e.,

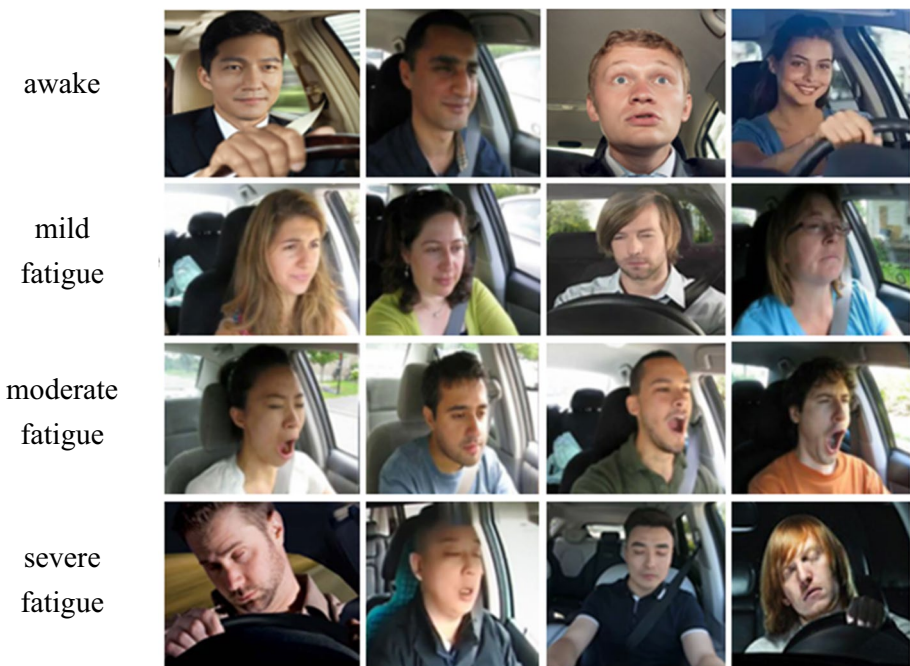


Fig. 11 Partial data set

the time consumed by a single identification process. This aspect is essential because the fatigue detection system needs to determine the driver's state in real time.

Table 5 shows the results of the above models on the dataset. A0 indicates the average detection accuracy of the awake state, A1 indicates the average detection accuracy of the mild fatigue state, A2 indicates the average detection accuracy of the moderate fatigue state, A3 indicates the average detection accuracy of the severe fatigue state, and Av indicates the average detection accuracy of the four states.

The test results show that the algorithm proposed in this paper achieves a high detection accuracy of 90.34%, 93.17%, 95.46% and 99.67% for different fatigue states. The average accuracy is as high as 94.66%, which is 3.86% higher than that of the traditional RF model, 5% higher than that of the SVM, RBF and BC models, respectively. In addition, the algorithm runs relatively fast due to its optimization and lightweight design in each detection step and multi-feature parallel detection is employed to improve the computational efficiency. The test results are shown in Fig. 12, where the distinction between the four states can be clearly seen.

4.3 Validation test

For safety reasons, the fatigue state was manually simulated. The data set consisted of 900 sober driving samples (including 150 interference samples such as talking or rubbing eyes), 650 mild fatigue samples, 455 moderate fatigue samples, and 550 severe fatigue samples, for a total of 2555 valid samples. Each fatigue sample lasted between 3 and 8 min. Using artificial fatigue state simulations, the proposed algorithm was comprehensively evaluated for accuracy, as shown in Table 6.

The test results indicate that the algorithm has a high degree of accuracy in detecting fatigued driving behavior, with an average detection accuracy of 98.35%. However, the tests also revealed that the algorithm had errors and missed detections in all the test videos, which could be attributed to the duration and severity of fatigue.

To further verify the detection performance of the algorithm in this paper, it is compared with the current mainstream fatigue driving detection algorithms on the self-curated dataset, and the experimental results are shown in Table 7. From the above table, it can be seen that under lower computing power and lighter weight, the algorithm in this paper has the highest mean average accuracy, and the mAP is 1.6% higher than that of the lighter weight Efficient Det-D2, and the number of parameters of the model is lower, and the

Table 5 Test Results

Model	Accuracy/%					Time consumption /ms
	A0	A1	A2	A3	Av	
SVM	85.64	87.75	90.43	94.84	89.67	35
RBF	85.62	87.23	91.51	95.56	89.98	97
BC	86.07	88.31	90.11	94.18	89.67	38
RF	86.25	88.40	91.64	96.94	90.80	39
RFWF	87.34	89.18	93.52	99.66	92.42	40
The proposed algorithm	90.34	93.17	95.46	99.67	94.66	37

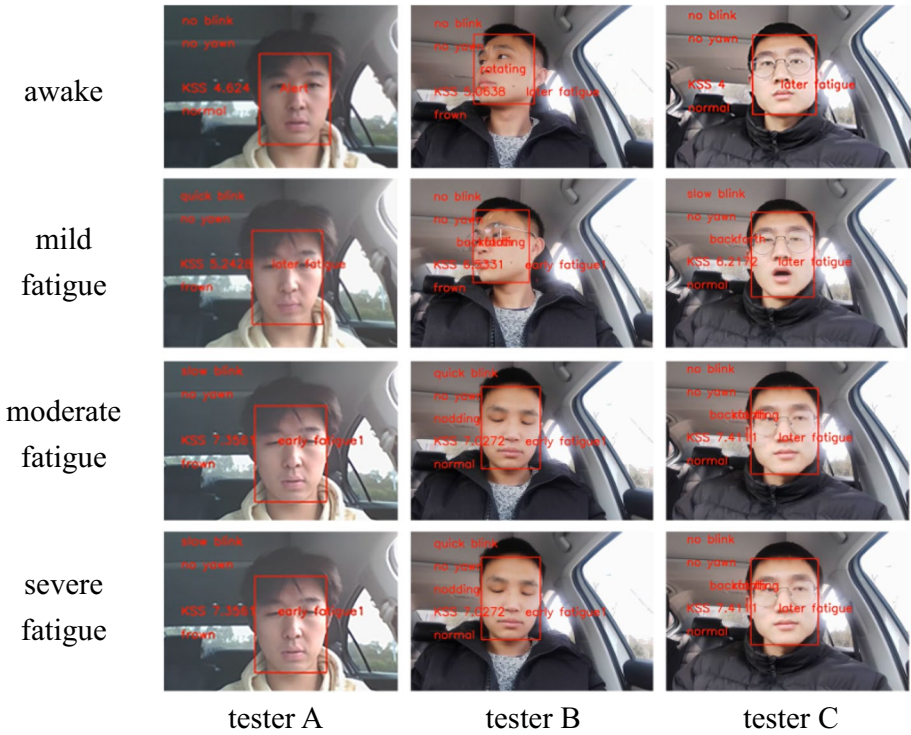


Fig. 12 Detection status diagram

Table 6 Comprehensive evaluation of the proposed algorithm

Video	Total Frames	Correct	Error	Undetected	Accuracy /%
A	240	230	1	9	95.93
B	240	239	1	0	99.58
C	240	238	0	2	99.16
D	240	237	3	0	98.75

Table 7 Validation of the mainstream algorithms on a self-curated dataset

Detection Methods	mAP /%	GFLOPs	Paras / ($\times 10^6$)	Vol/MB
YOLOv5-Tiny	89.9	12.9	8.68	66.7
YOLOv7	95.6	154.6	61.52	123.5
SSD	97	61.2	20.4	90.7
YOLOv8	96.1	160	69.36	161.1
MTCNN [35]	89.36	-	-	-
Dlib [36]	96.8	-	-	-
Bi-LSTM [37]	89.2	-	-	-
EfficientDet-D2	97.3	11	8.1	35.1
YOLOv9s	97.1	15.8	7.02	14.4
The proposed algorithm	98.9	4.5	3.32	7

computational complexity is smaller. This is because this paper comprehensively considers the lightweight processing and the depth extraction of face information, and further strengthens the focus on category features and the connection of contextual information by constructing feature mapping and lightweight feature enhancement module. In summary, the algorithm in this paper has strong comprehensive detection performance.

5 Conclusions

This paper presents a comprehensive facial feature-based driver fatigue detection algorithm that integrates several innovative techniques to improve detection accuracy and reliability. The main features of the proposed algorithm are:

1. The multi-feature fusion approach not only detects typical fatigue indicators such as blinking and yawning, but also incorporates new fatigue indicators such as head tilts forward and backward, thereby improving the overall comprehensiveness and precision of detection.
2. By fusing and analyzing multiple fatigue-related features, the algorithm can more accurately detect a range of driver postures, resulting in improved overall detection accuracy and robustness.
3. The algorithm's ability to map facial movements to KSS scores enables real-time assessment of fatigue levels, improving the system's performance and accuracy in detecting driver drowsiness.
4. The approach of decomposing fatigue videos into long and short KSS sequences, followed by early and late machine learning training, allows the algorithm to more effectively utilize the available training data, thereby improving its generalization ability and adaptability.

The proposed algorithm can effectively detect driver fatigue and provide timely warning signals, which is significant for promoting traffic safety and provides valuable insights for the future development of fatigue detection technology.

Funding This work was supported by the National Natural Science Foundation of China (51875494) and Yancheng Institute of Technology Graduate Innovation Program (SJCX22_XY045).

Data availability Not applicable.

Declarations

Conflict of interests The authors declare that we have no conflict of interest.

References

1. Yan M, Chen W, Wang J, Zhang M, Zhao L (2021) Characteristics and causes of particularly major road traffic accidents involving commercial vehicles in China. *Int J Environ Res Public Health* 18(8):3878. <https://doi.org/10.3390/ijerph18083878>
2. Vanlaar W, Simpson H, Mayhew D, Robertson R (2008) Fatigued and drowsy driving: a survey of attitudes, opinions and behaviors. *J Safety Res* 39(3):303–309. <https://doi.org/10.1016/j.jsr.2007.12.007>

3. Zhao C, Zhao M, Liu J, Zheng C (2012) Electroencephalogram and electrocardiograph assessment of mental fatigue in a driving simulator. *Accid Anal Prev* 45:83–90. <https://doi.org/10.1016/j.aap.2011.11.019>
4. Chowdhury A, Shankaran R, Kavakli M, Haque MM (2018) Sensor applications and physiological features in drivers' drowsiness detection: A review. *IEEE Sens J* 18(8):3055–3067. <https://doi.org/10.1109/JSEN.2018.2807245>
5. Hu X, Lodewijks G (2020) Detecting fatigue in car drivers and aircraft pilots by using non-invasive measures: The value of differentiation of sleepiness and mental fatigue. *J Safety Res* 72:173–187. <https://doi.org/10.1016/j.jsr.2019.12.015>
6. Chen X, Sun J, Ma Z, Sun J, Zheng Z (2020) Investigating the long- and short-term driving characteristics and incorporating them into car-following models. *Transport Res C-Emer* 117:102698. <https://doi.org/10.1016/j.trc.2020.102698>
7. Brown J, Jonsson M (2021) Deep learning for driver sleepiness classification using bioelectrical signals and Karolinska sleepiness scale. Master's Thesis. Linköping University. <https://www.diva-portal.org/smash/get/diva2:1582032/FULLTEXT01.pdf>. Accessed 2024-07-10
8. VenkataPhanikrishna B, Jaya Prakash A, Suchismitha C (2021) Deep review of machine learning techniques on detection of drowsiness using EEG signal. *IETE J Res* 69(6):3104–3119. <https://doi.org/10.1080/03772063.2021.1913070>
9. Mbouna RO, Kong SG, Chun M-G (2013) Visual analysis of eye state and head pose for driver alertness monitoring. *IEEE T Intell Transp* 14(3):1462–1469. <https://doi.org/10.1109/TITS.2013.2262098>
10. Ahmed S, Lee Y, Lim YH, Cho S-H, Park HK, Cho SH (2022) Noncontact assessment for fatigue based on heart rate variability using IR-UWB radar. *Sci Rep* 12:14211. <https://doi.org/10.1038/s41598-022-18498-w>
11. Fan C, Huang S, Lin S, Xu D, Peng Y, Yi S (2022) Types, risk factors, consequences, and detection methods of train driver fatigue and distraction. *Comput Intell Neurosci* 2022:8328077. <https://doi.org/10.1155/2022/8328077>
12. Lee H, Lee J, Shin M (2019) Using wearable ECG/PPG sensors for driver drowsiness detection based on distinguishable pattern of recurrence plots. *Electronics* 8(2):192. <https://doi.org/10.3390/electronic8020192>
13. Pankaj Kumar A, Komaragiri R, Kumar M (2022) A review on computation methods used in photoplethysmography signal analysis for heart rate estimation. *Arch Computat Methods Eng* 29:921–940. <https://doi.org/10.1007/s11831-021-09597-4>
14. Jia H, Xiao Z, Ji P (2022) End-to-end fatigue driving EEG signal detection model based on improved temporal-graph convolution network. *Comput Biol Med* 152:106431. <https://doi.org/10.1016/j.compbiomed.2022.106431>
15. Heydari N, Jiang N (2023) Heartbeat detection from single-lead ECG contaminated with simulated EMG at different intensity levels: A comparative study. *Biomed Signal Proces* 83:104612. <https://doi.org/10.1016/j.bspc.2023.104612>
16. Xu J, Li X, Chang H, Zhao B, Tan X, Yang Y, Tian H, Zhang S, Ren TL (2022) Electrooculography and tactile perception collaborative interface for 3D human-machine interaction. *ACS Nano* 16(4):6687–6699. <https://doi.org/10.1021/acsnano.2c01310>
17. Liu F, Chen D, Zhou J, Xu F (2022) A review of driver fatigue detection and its advances on the use of RGB-D camera and deep learning. *Eng Appl Artif Intel* 116:105399. <https://doi.org/10.1016/j.engappai.2022.105399>
18. Lampe D, Deml B (2022) Reducing passive driver fatigue through a suitable secondary motor task by means of an interactive seating system. *Appl Ergon* 103:103773. <https://doi.org/10.1016/j.apergo.2022.103773>
19. Parsa MJ, Javadi M, Mazinan AH (2022) Fatigue level detection using multivariate autoregressive exogenous nonlinear modeling based on driver body pressure distribution. *P I Mech Eng D-J Aut* 236(1):168–184. <https://doi.org/10.1177/09544070211014290>
20. Zhu T, Zhang C, Wu T, Ouyang Z, Li H, Na X, Liang J, Li W (2022) Research on a real-time driver fatigue detection algorithm based on facial video sequences. *Appl Sci-Basel* 12(4):2224. <https://doi.org/10.3390/app12042224>
21. Zhao G, He Y, Yang H, Tao Y (2022) Research on fatigue detection based on visual features. *IET Image Process* 16(4):1044–1053. <https://doi.org/10.1049/ipr2.12207>
22. Akrouf B, Mahdi W (2021) A novel approach for driver fatigue detection based on visual characteristics analysis. *J Amb Intel Hum Comp* 14(1):527–552. <https://doi.org/10.1007/s12652-021-03311-9>
23. Zhao C, Lian J, Dang Q, Tong C (2014) Classification of driver fatigue expressions by combined curvelet features and Gabor features, and random subspace ensembles of support vector machines. *J Intell Fuzzy Syst* 26(1):91–100. <https://doi.org/10.3233/IFS-120717>

24. Ahmad R, Borole JN (2015) Drowsy driver identification using eye blink detection. *Int J Comput Sci Inf Technol* 6(1):270–274. <https://www.ijcsit.com/docs/Volume%206/vol6issue01/ijcsit2015060161.pdf>. Accessed 10 July 2024
25. Ghoddoosian R, Galib M, Athitsos V (2019) A realistic dataset and baseline temporal model for early drowsiness detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp 178–187. <https://doi.org/10.1109/CVPRW.2019.00027>
26. Li K, Gong Y, Ren Z (2020) A fatigue driving detection algorithm based on facial multi-feature fusion. *IEEE Access* 8:101244–101259. <https://doi.org/10.1109/ACCESS.2020.2998363>
27. Ganesan S, Manousakis JE, Mulhall MD, Sletten TL, Tucker A, Howard ME, Anderson C, Rajaratnam SMW (2022) Sleep, alertness and performance across a first and a second night shift in mining haul truck drivers. *Chronobiol Int* 39(6):769–780. <https://doi.org/10.1080/07420528.2022.2034838>
28. Guo J, Deng J, Lattas A, Zafeiriou S (2021) Sample and computation redistribution for efficient face detection. <https://doi.org/10.48550/arXiv.2105.04714>
29. Mills KG, Niu D, Salameh M, Qiu W, Han F X, Liu P, Zhang J, Lu W, Jui S (2023) AIO-P: Expanding neural performance predictors beyond image classification. In: Proceedings of the AAAI Conference on Artificial Intelligence 37(8):9180–9189. <https://doi.org/10.1609/aaai.v37i8.26101>
30. Howard, Andrew G. et al (2019) Searching for MobileNetV3. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp 1314–1324. <https://doi.org/10.1109/ICCV.2019.00140>
31. Sri Mounika TV, Phanindra PH, Sai Charan NV, Kranthi Kumar Reddy Y, Govindu S (2022) Driver drowsiness detection using eye aspect ratio (EAR), mouth aspect ratio (MAR), and driver distraction using head pose estimation. In: *ICT Systems and Sustainability*, pp 619–627. https://doi.org/10.1007/978-981-16-5987-4_63
32. Dewi C, Chen RC, Jiang X et al (2022) Adjusting eye aspect ratio for strong eye blink detection based on facial landmarks. *PeerJ Comput Sci* 8:e943. <https://doi.org/10.7717/peerj-cs.943>
33. Janani RP, Narayanan KL, Krishnan RS, Kannan P, Kabilan R, Muthukumaran N (2022) Intelligent drowsiness and illness detection assist system for drivers. In: 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS). IEEE, pp 1150–1155. <https://doi.org/10.1109/ICAIS53314.2022.9743075>
34. Jahan I, Uddin KM, Murad SA, Miah MSU, Khan TZ, Masud M, Aljahdali S, Bairagi AK (2023) 4D: a real-time driver drowsiness detector using deep learning. *Electronics* 12(1):235. <https://doi.org/10.3390/electronics12010235>
35. Wang P, Shin HL, Yin Y, Lyu H (2022) A detection algorithm for the fatigue of ship officers based on deep learning technique. *Traffic Information and Safety* 40(1):63–71. <https://doi.org/10.3963/j.jssn.1674-4861.2022.01.008>
36. Liao DJ (2023) Yawn detection method based on dlib and variant transformer. *Automotive Technology* 3:42–48. <https://doi.org/10.19620/j.cnki.1000-3703.20220453>
37. Gao QH, Xie K, He ZF et al (2023) Fatigue driving detection by multi-modal feature fusion in complex environment. *Electron Meas Technol* 46(6):106–115. <https://doi.org/10.19651/j.cnki.emt.2210750>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.